Automatic tracking and control for web recommendation New approaches for web recommendation

Samuel Nowakowski, Anne Boyer

Automatic tracking and control for web recommendation

New approaches for web recommendation

Samuel Nowakowski , Anne Boyer
LORIA-KIWI
Campus scientifique – BP239 - F-54506 Vandœuvre-lès-Nancy Cedex
Samuel.nowakowski@loria.fr
Anne.boyer@loria.fr

***Abstract*— Recommender systems provide users with pertinent resources according to their context and their profiles, by applying statistical and knowledge discovery techniques. This paper describes a new approach of generating suitable recommendations based on the active user's navigation stream, by considering long distance resources in the history. Our main idea to solve this problem is the following: we consider that users browsing web pages or web contents can be seen as objects moving along trajectories in the web space. Having this assumption, we derive the appropriate description of the so-called recommender space to propose a mathematical model describing the behavior of the users/targets in the web/along the trajectories inside the recommender space.**

The second main assumption can then be expressed as follow: if we are able to track the users/targets along their trajectories, we are able to predict the future positions in the sub-spaces of the recommender space i.e., we are able to derive a new method for web recommendation and behavior monitoring.

To achieve these objectives, we use the theory of the dynamic state estimation and more specifically the theory of Kalman filtering. We establish the appropriate model of the target tracker and we derive the iterative formulation of the filter. Then, we propose a new recommender system formulated as a control loop. We validate our approach on data extracted from online video consumption and we derive a users monitoring approach.

Conclusions and perspectives are derived from the analysis of the obtained results and focus on the formulation of a topology of the recommender space.

Keywords: Recommender system; user profile; target tracking; Kalman filter;

I. INTRODUCTION

In Web-based services of dynamic content, recommender systems face the difficulty of identifying new pertinent items and providing pertinent and personalized recommendations for users.

Predicting the most pertinent resources for a given user is a key challenge in application areas such as e-commerce, resources access, web navigation and user interaction in online communities. One of the standard approaches used in this context is the Collaborative Filtering (CF) [12, 7]. CF systems automate the recommendation process based solely on user opinions, ignoring the content of resources. When asked for a recommendation, CF systems first identify users similar to the active user (her neighborhood) by using a similarity matrix and suggest the resources these users have rated highly in the past. Despite CF has achieved notable successes, several drawbacks remain: most of the time, the task of finding similar users is performed online, resulting in latency and a lack of scalability ([21], [19]). Moreover, in traditional CF the context of the recommendation is not considered, all ratings of the active user and her neighborhood are considered, whatever is the context of these ratings.

Exploiting the context of the active user is an important point of view to improve accuracy of recommender systems. Interests of users may indeed evolve according to the period of the day, their humor or past actions. In the frame of Web navigation for example, considering the context is appropriate [18] and can provide more accurate predictions. Scalability of recommendation systems is also crucial as current applications deal with more and more resources and the access to information gets democratized. We propose here a recommender system that considers the context of the recommendation. However, modeling and predicting users behavior in context involves generally a complex model, the challenge being to find a low enough complexity model that integrates context while providing a high accuracy.

The rest of this paper is organized as follows. We are first interested in works related to context-dependent recommendations. Then the similarities between target tracking and Web navigation are put forward. Second, we address the general issue of applying statistical language models to make recommendations. Based on this discussion, we define our skipping based recommender. The following section is dedicated to the presentation of the way we evaluated our model and the results obtained. Conclusion and perspectives are put forward in the last section.

The main idea of this paper is to propose an alternative way for recommender systems. Our work is based on the following assumption: we consider Users as target moving along a trajectory in the recommender space. This dynamic system can be modeled by techniques coming from control system methods and we use Kalman filtering to predict future positions of the users in the recommender space i.e.

the expectable movies categories to be seen. We will detail the backgrounds of this approach. Then, we expose the recommendation strategies. Our conclusion will give some guidelines for future works.

II. Related research

As discussed in the introduction, our primary focus in this section is context dependent recommendation systems. In contrast to collaborative filtering, the use of data mining techniques enables offline pattern discovery. Recommendation systems based on the use of data mining techniques are therefore scalable while handling the context.

Two main approaches of contexts are considered in the literature, the first one considers the context as a set of non-sequential resources, whereas the second approach represents the context as an ordered set of resources. Both these approaches use machine learning algorithms to discover patterns that are used to predict users future interests. Before presenting pattern discovery algorithms, we are first interested in the presentation of the data used.

III. Input Data

The input data we consider is made up of traces of usage (log files and users' ratings) of users interacting with the system (intranet, web site, etc.). This data is first used to discover patterns of usage then to perform recommendations.

IV. Markov model

One well-known approach to exploit user's history is to compute predictions by using Markov models. The use of Markov models in the frame of the web has been first dedicated to the reducing of access time by pre-fetching and caching pages [15]. With the same goal, [4] estimated conditional probabilities of transitioning directly from one page to another page within a time. First order Markov models are not very accurate in predicting the user's browsing behavior since these models do not look far in the past to efficiently discriminate the different histories [6]. [17, 18] showed that the prediction accuracy is increased when using a longer history. Higher order Markov models are used to capture longer histories, these are called k^{th} order Markov models. Given the navigation history of size k, the probability of each resource is computed, the resources with the highest conditional probability will be recommended. The use of k^{th} order Markov models lead to a high accuracy.

Let us notice that k^{th} order Markov models are similar to frequent contiguous patterns of fixed size $k + 1$ in the case the support and confidence thresholds are set to 0. One drawback of k^{th} order Markov models is the storage requirements, indeed in a k^{th} order Markov model a huge number of states are handled (this number increases according to the order of the model) [6]. Moreover, as with previous approaches, we are faced to a reduced coverage due to the problem of matching the active history and training data. As pruning is rarely done in Markov models, this coverage problem is however reduced. Many approaches can overcome coverage limitation. For example, we can mention the development of Markov models of orders varying from 1 to k called the all k^{th} order Markov model [14]. However such a model dramatically increases the complexity and storage space drawback. Many works have been led to improve this previous model.

When using Markov models, the order of navigation is taken into account and the sequences are strictly contiguous, these models are thus not permissive. If a given user makes parallel navigations or goes to an unwanted resource (noise), the model cannot correctly handle such a behavior and will thus reduce the size of the history considered. Such situations are handled by association rules and sequential patterns as resources are not contiguous. Moreover, when the model does not match the complete history, the most distant consulted resources are discarded for computing predictions. Thus, the most recent resources are always considered while some of them may be not important or may be navigation mistakes and should be discarded.

In the following section, we show how we can derive from Markov model an approach based on Kalman filtering and target tracking. In our approach, we based our recommendation strategy on a transformation of the web space.

V. Principles

Kalman filter is an optimal state estimator of a linear system. It can estimate the state of the system using a priori knowledge of the evolution of the state and the measurements. Kalman filter has main applications control systems and in target tracking.

A. Target tracking in the cyberspace

Figure 1 shows the principle of our approach. We consider a user which browses web pages or online resources (video, music, …). Each page/resource belongs to a category (categories are related to the classification of the available resources). All possible categories will define the geometrical structure of the space and the vector associated to one seen page will be as described in Figure 1.

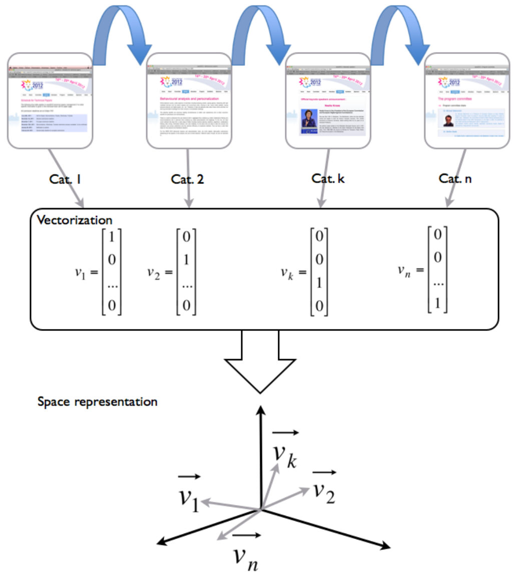

Figure 1. Principle

Successive vectors ($\overrightarrow{v_1}$ to $\overrightarrow{v_n}$) give successive "positions" in the space. Then these "positions" define the trajectory of the user in the recommender space. Each vector has the following dimensions:

$m \times 1$, m is the number of categories; each seen page belongs to one or more categories. In our case, each page coming from the tested site is classified (news, politics, education, …). The structure of each vector will be as follow:

$$\begin{bmatrix} cat-1 \\ cat-2 \\ \dots \\ cat-i \\ \dots \\ cat-m \end{bmatrix} \leftarrow corresponding \quad category \begin{bmatrix} 0 \\ 0 \\ \dots \\ 1 \\ \dots \\ 0 \end{bmatrix} = \overrightarrow{v_k} \tag{1}$$

Each vector of position could be multiplied by a positive scalar corresponding to the rating given by the user. For example, if someone (user j) has seen the resources corresponding to the i^{th} category, and has given the rate R_j, the vector of position will be as follows:

$$\overrightarrow{v_k} = \begin{bmatrix} 0 \\ 0 \\ \dots \\ R_j \\ \dots \\ 0 \end{bmatrix}$$

Our hypothesis: the user is then represented as a target whose positions are given by the successive vectors. We can model the trajectory and by using a state space cinematic model, we can predict future positions in the space.

B. *Kalman filter: equations*

1) *Hypothesis*

Our second hypothesis is the following: considering that users are moving along a trajectory defined by a set of vectors, we assume that the user can be considered as a target which is described by three components in the state space i.e., position, speed and acceleration. These three components will completely describe the dynamics of the moving users ([1], [5], [8]).

Thus, we choose to represent the state vector by concatenating these three components. The state vector has the following form:

$$X_k = \begin{bmatrix} x \\ v \\ \gamma \end{bmatrix}_k \tag{2}$$

where: $\dim(X_k) = 3m \times 1$

- x contains the components of the position vector, dimensions $m \times 1$
- v contains the components of the speed vector, dimensions $m \times 1$
- γ contains the components of the acceleration vector, dimensions $m \times 1$.

The dynamic of this state vector will be modeled by a state space model which has the following form:

$$\begin{cases} X_{k+1} = AX_k + w_k \\ Z_k = HX_k + v_k \end{cases} \tag{3}$$

where matrix A includes the relationship between the position, its first and second derivations which will inform us on the geometrical characteristics of the trajectory. T is a parameter which introduces time in the equation. In our case, we consider T equal to 1 because time is fixed each time the user goes to another webpage. The results of the algorithm are not sensitive to T.

$$A = \begin{bmatrix} \alpha & T & \frac{1}{2}T^2 \\ 0 & \alpha & T \\ 0 & 0 & \alpha \end{bmatrix} \tag{4}$$

Where: $\dim(A) = 3m \times 3m$.

Many values of parameter α have been tested. The chosen value does not influence our numerical results.

w_k and v_k are random noises (their properties will be given in the next section) which takes into account unexpected variations in the trajectories.

Matrix H, called the measurement matrix, is structured to obtain the values of the positions in the recommender space. Thus, H will have the following structure:

$$H = \begin{bmatrix} 1 & 0 & .. & 0 & 0 & 0 & .. & 0 & 0 & .. & 0 & 0 \\ 0 & 1 & .. & 0 & 0 & 0 & .. & 0 & 0 & .. & 0 & 0 \\ .. & .. & 1 & 0 & 0 & .. & .. & .. & .. & .. & .. & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & .. & 0 & 0 & .. & 0 & 0 \end{bmatrix} \qquad (5)$$

Where: $\dim(H) = m \times 3m$

2) General equations of the filter

Having the state space model (equations 3) and the structure of the state vector, we can derive the equations of the filter. First, we give some important properties of the Kalman filter.

- Information about X and Z is given as a Markov model i.e., Z is a linear combination of the components of X;
- Estimations of X are obtained from any initial instant;
- Estimations can be obtained for non-stationary process i.e., time-varying models.
- w_k and v_k are uncorrelated white noises where $w_k \approx N(0,Q)$and $v_k \approx N(0,R)$. In our case, Q and R are taken to identity matrix.

The Kalman filter equations are then given by the following equations [8]:

Prediction: it is the predicted state knowing past values

$$\begin{cases} \hat{X}_{k+1/k} = \hat{X}_{k/k-1} + K_k\left(Z_k - H\hat{X}_{k/k-1}\right) \\ \qquad\quad = (A - K_k H)\hat{X}_{k/k-1} + K_k Z_k \end{cases} \qquad (6)$$

Kalman gain: it described the dynamic of the filter. The dynamic takes into account the variations of the moving target.

$$K_k = AP_{k/k-1}H^T\left(HP_{k/k-1}H^T + R\right)^{-1} \qquad (7)$$

The evolution of the uncertainty on the estimation is then given by the following Riccati equation:

$$P_{k+1/k} = AP_{k/k-\&}A^T - AP_{k/k-1}H^T\left(HP_{k/k-1}H^T + R\right)^{-1}HP_{k/k-1}A^T \qquad (8)$$

where the initial conditions (which initialize the filter) are given by:

$$\hat{X}_{0/-1} = X_0, \; P_{0/-1} = P_0 \qquad (9)$$

and the state prediction is given by: $\hat{X}_{k+1/k}$

The state prediction will predict the future position of the user in the state space.

Because of the structure of the vectors, having 0 when the web pages does not belong to the corresponding category and 1 when it belongs to it, the trajectory will be on a hyper-sphere as shown in the following figure (red arrows show the movement from page to page):

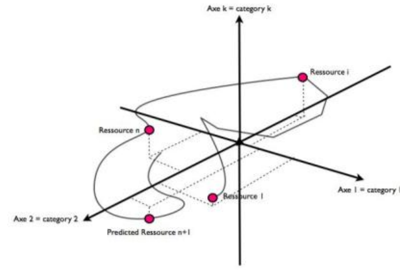

Figure 2. Trajectory in the recommender space

The Kalman predictor will predict the future position in the recommender space i.e. the most possible category knowing the past of the user.

VI. RECOMMENDATION STRATEGY

In this approach, we can build a recommendation by analyzing the prediction provided by Kalman filter.

A. Principle

The profile is built from list of pages seen on the studied website. Each page/resource is defined by a subset of categories such as “Drama”, “Entertainment”, “Adventure”, etc.

Our new recommending strategy is based on control loop which can be described by Figure 3.

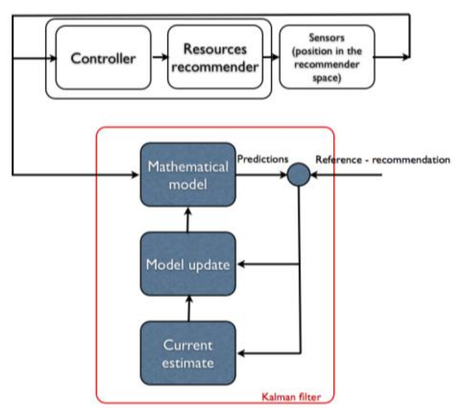

Figure 3. Control loop for recommendation

This control loop will observe the difference between the estimated value of the category and the calculated category and it will integrate the controller/recommender to build the most accurate model of the user. Having this, this configuration can predict where in the recommender space, the user will “move”. The recommendation strategy will use the predicted position to “suggest” to the user the appropriate category of contents.

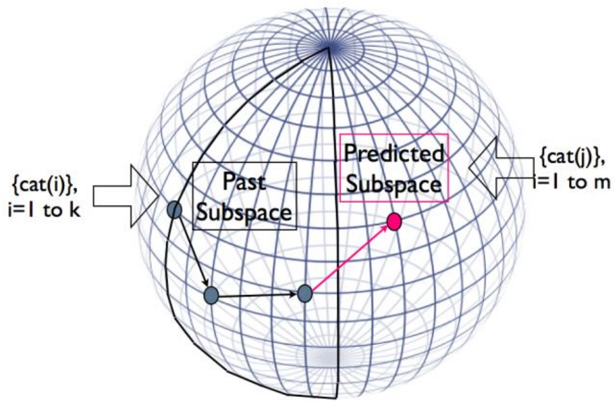

Figure 4. Trajectory and predicted "positions"

Conversely to existing methods which recommend precise contents for a given user, this method will performs on the macroscopic level, i.e., subspaces of specific categories. The strategy will isolate the appropriate subspace and the recommendation will be done in the related categories. Then, we can imagine that we will have a more precise recommendation by computing a second iteration (target tracking in the trajectory in the subspace and positions prediction) on the subspace (zoom effect).

To summarize, the recommendation is based on the two preceding arguments.

- the user's actual state of mind
- the subset of retained dimensions.

We then have to define the recommendation for a set of contents. Furthermore, according to what the user watched during the day, we can refine our recommendation. Indeed, in our example, if the user is interested in contents of types x, y and z and if he has already watched content of type x and y that day, the recommendation would essentially concentrate on content of type z.

Hence, we will need to make a last step which will be devoted to the identification of the appropriate content which corresponds to the estimation of the dimensions' evolution.

B. The experiment

1) Description of the experiment

The dataset is the TV consumption of 6423 English households over a period of 6 months (from 1st September 2008 to 1st March 2009) (Broadcaster Audience Research Board, [2]), [3]. This dataset contains information about the user, the household and about television program. Each TV program is labeled by one or several genres. In the experiment, a user profile build for each person. The user profile is the set of genres associated to the value of interest of the user for each genre. This user profile is elaborated in function of the quality of a user's TV consumption: if a TV program is watched entirely, the genre associated to this TV program increases in the user profile. Several logical rules are applied to estimate the interest of a user for a TV program.

The methodology of the experimentation is the following: The Kalman filter is applied iteratively to estimate the future positions of the user in the space.

The entire consumption is described by 44 categories corresponding to the 44 dimensions of the recommender space where users are "moving".

2) Numerical results

The obtained results can be exposed as follows: Kalman filter predicts the interest of a specific user for a subset of categories knowing his past.

Using this prediction, we can propose a new recommendation strategy:

- If the Quantity of Interest (QoI) of the user is predicted to be in one specific region of the space, we can recommend something inside this specific region:
- For example, if the specific region is defined by dimensions Documentary and Drama, we can recommend contents related to these two dimensions
- If the predicted quantity of interest (QoI) changes to another dimension of the space, we can automatically recommend content from this new region of the space.

3) Results

The results can be analyzed as follows: Kalman filter predicts the specific interest for a category of contents of one user.

Figures 5 and 6 show Estimation / Prediction computed by Kalman filtering. Doted-lines show the evolution of the real values. Continuous lines show the obtained predictions. We can see the estimation/prediction given by the Kalman filter: green lines show the prediction obtained at each time using the knowledge we have of the degree of interest of each user. We can see that the prediction fits the real values even if they present abrupt variations.

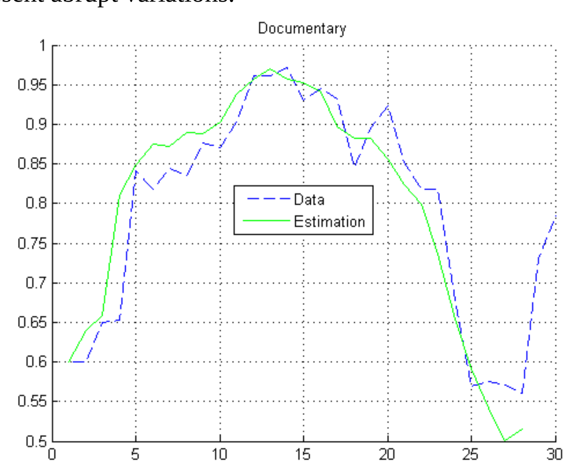

Figure 5. Prediction for Drama

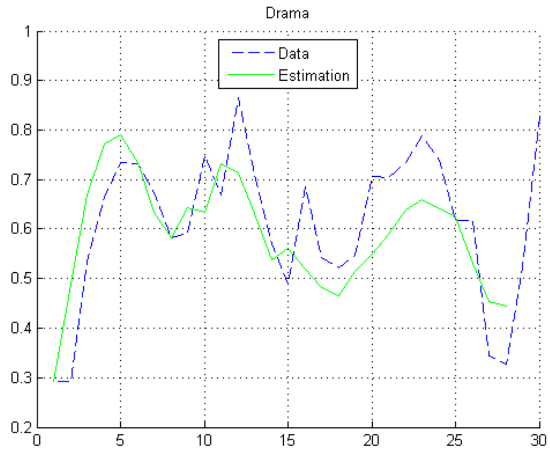

Figure 6. Prediction for Documentary

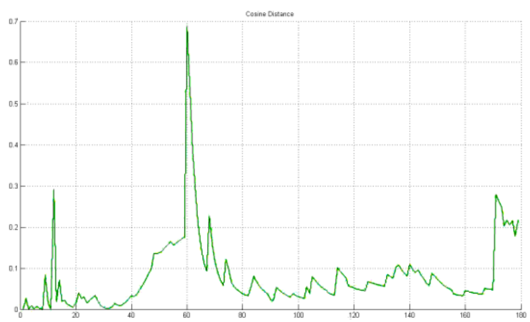

Figure 7. Cosine distance

Figure 7 shows the results of the cosine distance that has been computed between the true values and the prediction by the filter. It shows that the prediction will quickly converge with the true values.

In this approach, we can build a recommendation by analyzing the estimation provided by Kalman filter.

The profile is built from the consumption of TV programs. Each TV program is defined by concepts such as entertainment, science fiction, talk show, etc. The analysis of the way different TV programs are watched allows deducing the interest of a user for each concept.

Our new recommending strategy can be derived as follows:

- If the computed concept is superior to the estimated concept (noted negative difference), then the user's interest for this concept is decreasing.
- If the estimated concept is superior to the computed concept (noted positive difference), then the user's interest for this concept is increasing.

The process will focus on the concepts showing up a big difference: the concepts with an important positive difference influence the recommendation towards these concepts, whereas the concepts with an important negative difference discourage the recommendation towards these concepts.

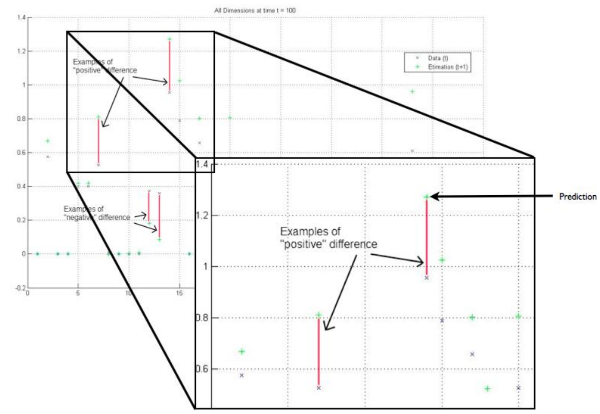

Figure 8. Analysis of the evolution of the prediction for recommendation

Conversely to existing methods that recommend precise contents for a given user, this method will performs on the macroscopic level i.e. subspaces of specific categories. The strategy will isolate the appropriate subspace and the recommendation will be done in the related categories. Then, we can imagine that we have a more precise recommendation by computing a second iteration (target tracking in the trajectory in the subspace and positions prediction) on the subspace (zoom effect).

To summarize, the recommendation is based on the two preceding arguments.

- the user's actual state of mind
- the subset of retained dimensions.

From these “positive” or “negative” dimensions and from the TV program, we have to define the recommendation for a set of TV programs for that day. Furthermore, according to what the user watched during the day, we can refine our recommendation. Indeed, in our example, if the user is interested in contents of types x, y and z and if he has already watched content of type x and y that day, the recommendation would essentially concentrate on content of type z.

Hence we need to make a last step to find a content that corresponds to the estimation of the dimensions' evolution.

VII. CONCLUSION

Our original approach considering users as target moving along trajectories in subspaces of the recommendation space will consider web recommendation as a control system problem. Web recommendation becomes a system described by a state space model to be controlled or tracked. By comparing inputs to predicted and/estimated values, we obtain a new kind of recommender systems that will consider as moving targets to be identified or dynamic systems to be controlled.

Then, in our case, knowing the past positions of the user in this space along the different axis of the 44 dimensions space, our Kalman filter based recommender system will suggest:

- if the user is interested in contents of types x, y and z and if he has already watched content of type x and y that day, the recommendation would essentially concentrate on content of type z
- knowing the position in the space, the best prediction for his next positions in the recommender space i.e. his best index of interest related to the favorite contents.

At last, the strength of our approach is in its capability to make recommendations at a higher level that fit users habits i.e. given main directions to follow knowing the trajectory in the space and not to suggest specific resources. Furthermore, the trajectories in the recommender space give the opportunity to compute a monitoring system where we can visualize in real time the users' trajectories (see Figure 9).

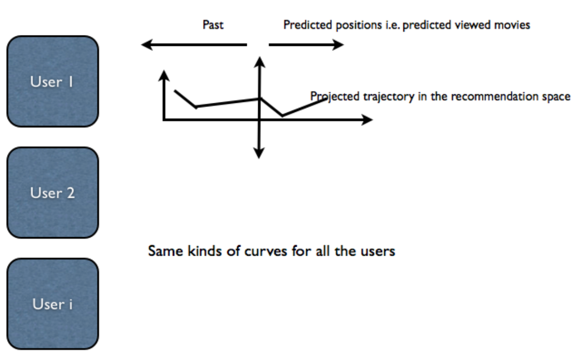

Figure 9. Users monitoring

Future works will be focused on tracking groups of users and on the definition of the topology of the recommendation space as a space including specific mathematical operators. Our results are also applied on data coming from news websites to predict the most accurate contents to be recommended to a set of users. Another in progress application concerns eLearning platforms and the pedagogical resources recommendation.

References

[1] B. Anderson, and J.B. Moore, 1977. Optimal filtering. Prentice Hall – Information and System Sciences Series

[2] BARB: Broadcaster Audience Research Board, DOI=http://www.barb.co.uk/.

[3] C. Bernier, A. Brun, A. Aghasaryan, M. Bouzid, J. Picault, C. Senot and A. Boyer, 2010. "Topology of communities for the collaborative recommendations to groups," in SIIE 2010, 2010.

[4] A. Bestravos, "Using Speculation to reduce Server Load and Service Time on the WWW" in *Conference on Information and Knowledge Management*, pp. 403–410, 1995.

[5] G.E.P. Box, and G.M. Jenkins, 1970. Time series analysis : forecasting and control. Holden Day.

[6] M. Despande and G. Karypis, "Selective Markov Models for Predicting Web Pages Accesses," *ACM Trans. Internet Technol.*, pp. 163-184, 2004.

[7] A. Das, M. Datar, A. Garg and S. Rajaram, "Google News personalization: scalable online collaborative filtering" in *WWW'07 Proc. Of the 16th International Conference on World Wide Web, 2007*, pp. 271–280

[8] Gevers, M. and Vandendorpe, L. Processus stochastiques, estimation et prediction. DOI=http://www.tele.ucl.ac.be/EDU/INMA2731/].

[9] W. Gibson, 1988. Neuromancien. Collection J'ai Lu

[10] D. Goldberg, D. Nichols, B. Oki and D. Terry. "Using Collaborative filtering to Weave an Information Tapestry" *Communication of the ACM*, pp. 61–70, 1992.

[11] D.Goldberg, D. Nichols, B. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of ACM, 1992*, pp. 61–70.

[12] D. Goldberg, D. Nichols, B. Oki, D. Terry, 1992. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, pages 61–70

[13] M. Grcar, D. Mladenic, B. Fortuna, M. Groblenik, 2006. Data Sparsity Issues in the Collaborative Filtering Framework. Advances in Web Mining and Web Usage Analysis, pages 58–76

[14] N. Nakagawa and B. Mobasher, "Impact of Site Characteristics on Recommendation Models based On association Rules and sequential patterns," in *Proceedings Of the IJCAI'03 Workshop on Intelligent Techniques for Web Personalization, 2003*.

[15] V. Padmanabhan and J. Mogul, "Using predictive prefetching to improve World Wide Web Latency," *Computer Communication Review*, pp. 22–36, 1996.

[16] M Pazzani, D. Billsus, 2007. Content-Based Recommendation Systems. The AdaptiveWeb, pages 325–341.

[17] P. Pirolli and J. Pitkow, "Distribution of Surfer's paths through the World Wide Web: Empirical Characterization," in *World Wide Web*, pp. 29–45, 1999.

[18] J. Pitkow and P. Pirolli, "Mining longest Repeating Subsequences to predict World Wide Web surfing." in *Proceedings Of the 2nd conference of USENIX Symposium on Internet Technologies and Systems, pp. 139-150, 1999.*

[19] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. "Analysis of recommender algorithms for e-Commerce," in *Proceedings Of the 2nd ACM e-Commerce conference, 2000*.

[20] T. Söderström, 1994. Discrete-time stochastic systems : estimation and control. Prentice Hall International.

[21] L.Ungar and D. Foster, "Clustering methods for collaborative filtering" in *Proceedings Of the AAI98 Workshop on Recommendation Systems, 1998*.